

In the Claims:

1. (Previously presented) A method of printing data from an application, comprising the steps of:
 - invoking a print function with a format argument that is a pointer to a memory location in an address space of the application and at least one data argument from the application;
 - saving the format argument and the at least one data argument in a deferred trace data buffer;
 - returning to the application that invoked the print function; then
 - processing the deferred trace data buffer to print the at least one data argument.
2. (Original) A method as recited in Claim 1, wherein the step of processing the deferred trace data buffer to print the at least one data argument comprises the steps of:
 - retrieving the format argument and the at least one data argument from the deferred trace data buffer;
 - formatting the at least one data argument based on the format argument; and
 - printing the formatted at least one data argument.
3. (Original) A method as recited in Claim 2, wherein the step of formatting the at least one data argument based on the format argument comprises the steps of:
 - determining if the format argument specifies a character string conversion; and
 - printing an address of a respective one of the at least one data argument that corresponds to the character string conversion.
4. (Original) A method as recited in Claim 1, further comprising the step of:
 - determining if a deferred print flag has been set.

5. (Original) A method as recited in Claim 4, wherein the step of saving the format argument and the at least one data argument in the deferred trace data buffer comprises the step of:

saving the at least one data argument in the deferred trace data buffer if the deferred print flag has been set; and

wherein the step of processing the deferred trace data buffer to print the at least one data argument comprises the step of:

processing the deferred trace data buffer to print the at least one data argument if the deferred print flag has been set.

6. (Original) A method as recited in Claim 1, wherein the step of saving the format argument and the at least one data argument in the deferred trace data buffer and the step of processing the deferred trace data buffer to print the at least one data argument are performed in different execution threads.

7. (Previously presented) A method as recited in Claim 1, further comprising the step of:

saving the deferred trace data buffer and a memory contents comprising the address space of the application in a non-volatile storage medium.

8. (Original) A method as recited in Claim 7, wherein the step of saving the format argument and the at least one data argument in the deferred trace data buffer is performed on a first computing machine and the step of processing the deferred trace data buffer to print the at least one data argument is performed on a second computing machine, the second computing machine being different from the first computing machine and having access to the address space of the application via the non-volatile storage medium.

9. (Previously presented) A method of printing data from an application, comprising the steps of:

invoking a print function with a format argument that is a pointer to a memory location in an address space of the application from the application;
saving the format argument in a deferred trace data buffer;
returning to the application that invoked the print function; then
processing the deferred trace data buffer to print the format argument.

10. (Previously presented) A method as recited in Claim 9, wherein the step of saving the format argument in the deferred trace data buffer comprises the step of:
saving the pointer in the deferred trace data buffer

11. (Original) A method as recited in Claim 10, wherein the step of processing the deferred trace data buffer to print the format argument comprises the step of:
processing the deferred trace data buffer to print a contents of the memory location in the address space of the application that is referenced by the pointer.

12. (Original) A method as recited in Claim 11, wherein the step of saving the pointer in the deferred trace data buffer and the step of processing the deferred trace data buffer to print the contents of the memory location in the address space of the application that is referenced by the pointer are performed in different execution threads.

13. (Original) A method as recited in Claim 11, further comprising the step of:
saving the deferred trace data buffer and a memory contents comprising the address space of the application in a non-volatile storage medium.

14. (Original) A method as recited in Claim 13, wherein the step of saving the pointer in the deferred trace data buffer is performed on a first computing machine and the step of processing the deferred trace data buffer to print the contents of the memory location in the address space of the application that is referenced by the pointer is performed on a second computing machine, the second computing machine being different from the first

computing machine and having access to the deferred trace data buffer and the address space of the application via the non-volatile storage medium.

15. (Previously presented) A method as recited in Claim 9, wherein the step of saving the format argument in the deferred trace data buffer comprises the step of:

saving a contents of the memory location in the address space of the application that is referenced by the pointer in the deferred trace data buffer.

16. (Original) A method as recited in Claim 15, wherein the step of processing the deferred trace data buffer to print the format argument comprises the step of:

processing the deferred trace data buffer to print the contents of the memory location in the address space of the application that is referenced by the pointer.

17. (Original) A method as recited in Claim 16, wherein the step of saving the contents of the memory location in the address space that is referenced by the pointer in the deferred trace data buffer and the step of processing the deferred trace data buffer to print the contents of the memory location in the address space of the application that is referenced by the pointer are performed in different execution threads.

18. (Original) A method as recited in Claim 16, further comprising the step of:
saving the deferred trace data buffer to a non-volatile storage medium.

19. (Previously presented) A method as recited in Claim 18, wherein the step of saving the contents of the memory location in the address space that is referenced by the pointer in the deferred trace data buffer is performed on a first computing machine and the step of processing the deferred trace data buffer to print the contents of the memory location in the address space of the application that is referenced by the pointer is performed on a second computing machine, the second computing machine being different from the first computing machine and having access to the deferred trace data buffer via the non-volatile storage medium.

20. (Previously presented) A system for printing data from an application, comprising:

means for invoking a print function with a format argument that is a pointer to a memory location in an address space of the application and at least one data argument from the application;

means for saving the format argument and the at least one data argument in a deferred trace data buffer;

means for returning to the application that invoked the print function; and

means for processing the deferred trace data buffer to print the at least one data argument after returning to the application that invoked the print function.

21. (Original) A system as recited in Claim 20, wherein the means for processing the deferred trace data buffer to print the at least one data argument comprises:

means for retrieving the format argument and the at least one data argument from the deferred trace data buffer;

means for formatting the at least one data argument based on the format argument;
and

means for printing the formatted at least one data argument.

22. (Original) A system as recited in Claim 21, wherein the means for formatting the at least one data argument based on the format argument comprises:

means for determining if the format argument specifies a character string conversion;
and

means for printing an address of a respective one of the at least one data argument that corresponds to the character string conversion.

23. (Original) A system as recited in Claim 20, further comprising:

means for determining if a deferred print flag has been set.

24. (Original) A system as recited in Claim 23, wherein the means for saving the format argument and the at least one data argument in the deferred trace data buffer comprises:

means for saving the at least one data argument in the deferred trace data buffer if the deferred print flag has been set; and
wherein the means for processing the deferred trace data buffer to print the at least one data argument comprises:

means for processing the deferred trace data buffer to print the at least one data argument if the deferred print flag has been set.

25. (Original) A system as recited in Claim 20, wherein the means for saving the format argument and the at least one data argument in the deferred trace data buffer and the means for processing the deferred trace data buffer to print the at least one data argument execute in different execution threads.

26. (Previously presented) A system as recited in Claim 20, further comprising:
means for saving the deferred trace data buffer and a memory contents comprising the address space of the application in a non-volatile storage medium.

27. (Original) A system as recited in Claim 26, wherein the means for saving the format argument and the at least one data argument in the deferred trace data buffer executes on a first computing machine and the means for processing the deferred trace data buffer to print the at least one data argument executes on a second computing machine, the second computing machine being different from the first computing machine and having access to the address space of the application via the non-volatile storage medium.

28. (Previously presented) A system for printing data from an application, comprising:

means for invoking a print function with a format argument that is a pointer to a memory location in an address space of the application from the application;

means for saving the format argument in a deferred trace data buffer;
means for returning to the application that invoked the print function; and
means for processing the deferred trace data buffer to print the format argument after returning to the application that invoked the print function.

29. (Previously presented) A system as recited in Claim 28, wherein the means for saving the format argument in the deferred trace data buffer comprises:

means for saving the pointer in the deferred trace data buffer.

30. (Original) A system as recited in Claim 29, wherein the means for processing the deferred trace data buffer to print the format argument comprises:

means for processing the deferred trace data buffer to print a contents of the memory location in the address space of the application that is referenced by the pointer.

31. (Original) A system as recited in Claim 30, wherein the means for saving the pointer in the deferred trace data buffer and the means for processing the deferred trace data buffer to print the contents of the memory location in the address space of the application that is referenced by the pointer execute in different execution threads.

32. (Original) A system as recited in Claim 30, further comprising:

means for saving the deferred trace data buffer and a memory contents comprising the address space of the application in a non-volatile storage medium.

33. (Original) A system as recited in Claim 32, wherein the means for saving the pointer in the deferred trace data buffer executes on a first computing machine and the means for processing the deferred trace data buffer to print the contents of the memory location in the address space of the application that is referenced by the pointer executes on a second computing machine, the second computing machine being different from the first computing machine and having access to the deferred trace data buffer and the address space of the application via the non-volatile storage medium.

34. (Previously presented) A system as recited in Claim 28, wherein the means for saving the format argument in the deferred trace data buffer comprises:

means for saving a contents of the memory location in the address space of the application that is referenced by the pointer in the deferred trace data buffer.

35. (Original) A system as recited in Claim 34, wherein the means for processing the deferred trace data buffer to print the format argument comprises:

means for processing the deferred trace data buffer to print the contents of the memory location in the address space of the application that is referenced by the pointer.

36. (Original) A system as recited in Claim 35, wherein the means for saving the contents of the memory location in the address space that is referenced by the pointer in the deferred trace data buffer and the means for processing the deferred trace data buffer to print the contents of the memory location in the address space of the application that is referenced by the pointer execute in different execution threads.

37. (Original) A system as recited in Claim 35, further comprising:

means for saving the deferred trace data buffer to a non-volatile storage medium.

38. (Previously presented) A system as recited in Claim 37, wherein the means for saving the contents of the memory location in the address space that is referenced by the pointer in the deferred trace data buffer executes on a first computing machine and the means for processing the deferred trace data buffer to print the contents of the memory location in the address space of the application that is referenced by the pointer executes on a second computing machine, the second computing machine being different from the first computing machine and having access to the deferred trace data buffer via the non-volatile storage medium.

39. (Previously presented) A computer program product for printing data from an application, comprising:

a computer readable storage medium having computer readable program code embodied therein, the computer readable program code comprising:

computer readable program code for invoking a print function with a format argument that is a pointer to a memory location in an address space of the application and at least one data argument from the application;

computer readable program code for saving the format argument and the at least one data argument in a deferred trace data buffer;

computer readable program code for returning to the application that invoked the print function; and

computer readable program code for processing the deferred trace data buffer to print the at least one data argument after returning to the application that invoked the print function.

40. (Original) A computer program product as recited in Claim 39, wherein the computer readable program code for processing the deferred trace data buffer to print the at least one data argument comprises:

computer readable program code for retrieving the format argument and the at least one data argument from the deferred trace data buffer;

computer readable program code for formatting the at least one data argument based on the format argument; and

computer readable program code for printing the formatted at least one data argument.

41. (Original) A computer program product as recited in Claim 40, wherein the computer readable program code for formatting the at least one data argument based on the format argument comprises:

computer readable program code for determining if the format argument specifies a character string conversion; and

computer readable program code for printing an address of a respective one of the at least one data argument that corresponds to the character string conversion.

42. (Original) A computer program product as recited in Claim 39, further comprising:

computer readable program code for determining if a deferred print flag has been set.

43. (Original) A computer program product as recited in Claim 42, wherein the computer readable program code for saving the format argument and the at least one data argument in the deferred trace data buffer comprises:

computer readable program code for saving the at least one data argument in the deferred trace data buffer if the deferred print flag has been set; and wherein the computer readable program code for processing the deferred trace data buffer to print the at least one data argument comprises:

computer readable program code for processing the deferred trace data buffer to print the at least one data argument if the deferred print flag has been set.

44. (Original) A computer program product as recited in Claim 39, wherein the computer readable program code for saving the format argument and the at least one data argument in the deferred trace data buffer and the computer readable program code for processing the deferred trace data buffer to print the at least one data argument execute in different execution threads.

45. (Previously presented) A computer program product as recited in Claim 39, further comprising:

computer readable program code for saving the deferred trace data buffer and a memory contents comprising the address space of the application in a non-volatile storage medium.

46. (Original) A computer program product as recited in Claim 45, wherein the computer readable program code for saving the format argument and the at least one data argument in the deferred trace data buffer executes on a first computing machine and the computer readable program code for processing the deferred trace data buffer to print the at least one data argument executes on a second computing machine, the second computing machine being different from the first computing machine and having access to the address space of the application via the non-volatile storage medium.

47. (Previously presented) A computer program product for printing data from an application, comprising:

a computer readable storage medium having computer readable program code embodied therein, the computer readable program code comprising:

computer readable program code for invoking a print function with a format argument that is a pointer to a memory location in an address space of the application from the application;

computer readable program code for saving the format argument in a deferred trace data buffer;

computer readable program code for returning to the application that invoked the print function; and

computer readable program code for processing the deferred trace data buffer to print the format argument after returning to the application that invoked the print function.

48. (Previously presented) A computer program product as recited in Claim 47, wherein the computer readable program code for saving the format argument in the deferred trace data buffer comprises:

computer readable program code for saving the pointer in the deferred trace data buffer.

49. (Original) A computer program product as recited in Claim 48, wherein the computer readable program code for processing the deferred trace data buffer to print the format argument comprises:

computer readable program code for processing the deferred trace data buffer to print a contents of the memory location in the address space of the application that is referenced by the pointer.

50. (Original) A computer program product as recited in Claim 49, wherein the computer readable program code for saving the pointer in the deferred trace data buffer and the computer readable program code for processing the deferred trace data buffer to print the contents of the memory location in the address space of the application that is referenced by the pointer execute in different execution threads.

51. (Original) A computer program product as recited in Claim 49, further comprising:

computer readable program code for saving the deferred trace data buffer and a memory contents comprising the address space of the application in a non-volatile storage medium.

52. (Original) A computer program product as recited in Claim 51, wherein the computer readable program code for saving the pointer in the deferred trace data buffer executes on a first computing machine and the computer readable program code for processing the deferred trace data buffer to print the contents of the memory location in the address space of the application that is referenced by the pointer executes on a second computing machine, the second computing machine being different from the first computing machine and having access to the deferred trace data buffer and the address space of the application via the non-volatile storage medium.

53. (Previously presented) A computer program product as recited in Claim 47, wherein the computer readable program code for saving the format argument in the deferred trace data buffer comprises:

computer readable program code for saving a contents of the memory location in the address space of the application that is referenced by the pointer in the deferred trace data buffer.

54. (Original) A computer program product as recited in Claim 53, wherein the computer readable program code for processing the deferred trace data buffer to print the format argument comprises:

computer readable program code for processing the deferred trace data buffer to print the contents of the memory location in the address space of the application that is referenced by the pointer.

55. (Original) A computer program product as recited in Claim 54, wherein the computer readable program code for saving the contents of the memory location in the address space that is referenced by the pointer in the deferred trace data buffer and the computer readable program code for processing the deferred trace data buffer to print the contents of the memory location in the address space of the application that is referenced by the pointer execute in different execution threads.

56. (Original) A computer program product as recited in Claim 54, further comprising:

computer readable program code for saving the deferred trace data buffer to a non-volatile storage medium.

57. (Previously presented) A computer program product as recited in Claim 56, wherein the computer readable program code for saving the contents of the memory location in the address space that is referenced by the pointer in the deferred trace data buffer executes on a first computing machine and the computer readable program code for processing the

In re: Fluke et al.
Serial No.: 09/607,074
Filed: June 29, 2000
Page 15 of 21

deferred trace data buffer to print the contents of the memory location in the address space of the application that is referenced by the pointer executes on a second computing machine, the second computing machine being different from the first computing machine and having access to the deferred trace data buffer via the non-volatile storage medium.